01/8036
ISR (2)

(19)日本国特許庁（ＪＰ）　(12)公開特許公報（Ａ）　(11)特許出願公開番号

特開平10－105472

(43)公開日　平成10年(1998)４月24日

(51)Int.Cl.[6]		識別記号	FI		
Ｇ０６Ｆ	12/14	３２０	Ｇ０６Ｆ	12/14	３２０Ｃ
		３１０			３１０Ｋ
Ｇ０６Ｋ	17/00		Ｇ０６Ｋ	17/00	Ｄ
	19/073			19/00	Ｐ

審査請求　未請求　請求項の数14　ＦＤ　（全 16 頁）

(21)出願番号　特願平8－278877

(22)出願日　平成８年(1996)９月30日

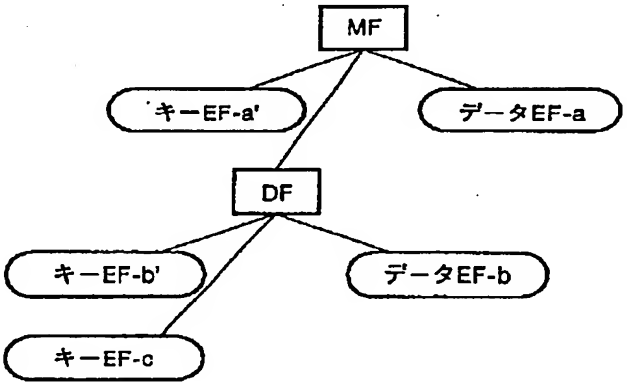
(71)出願人　000003078
株式会社東芝
神奈川県川崎市幸区堀川町72番地

(72)発明者　飯島　康雄
神奈川県川崎市幸区柳町70番地　株式会社東芝柳町工場内

(74)代理人　弁理士　鈴江　武彦　（外６名）

(54)【発明の名称】　メモリのアクセス管理方法

(57)【要約】
【課題】上位者からのデータファイルへの関与をなくし、上位者から下位者へ当該データファイルまたはメモリ領域へのアクセスの権限を委譲する。
【解決手段】メモリを複数のファイルに分割してなるＩＣカードにおいて、下位者は、上位者が設定したトランスポートキーを自身のみが知りえるキー（ｂ′）に当該キーＥＦに設定されているキー変更用のアクセス条件を参照して変更し、データファイル（ＤＦ）に付与されているトランスポートビットをオンにし、自身が必要とするデータＥＦ－ｂをデータファイル（ＤＦ）に設定されているＥＦ創成用アクセス条件を参照してデータファイル（ＤＦ）配下に創成し、自身が必要とするキーＥＦ－ｃをデータファイル（ＤＦ）に設定されているＥＦ創成用アクセス条件を参照してデータファイル（ＤＦ）配下に創成する。



1

【特許請求の範囲】

【請求項１】　メモリを複数のファイルに分割し、この分割した複数のファイルに対してのアクセスをそれぞれ管理するもので、当該メモリを用いたシステムの上位者にてファイル配下に第１のキーが予め設定されており、当該メモリを用いたシステムの下位者が新たにキーを設定可能に構成されており、かつ、これらのキーの照合によって上記ファイルへのアクセスを可能とするメモリのアクセス管理方法であって、

上記上位者が設定した第１のキーを、この第１のキーに設定されているキー変更用のアクセス条件を参照して下位者のみが知りえる第２のキーに変更し、この第２のキーに変更された際、上記上位者によるファイル配下のキー創成を拒絶するようにしたことを特徴とするメモリのアクセス管理方法。

【請求項２】　メモリを複数のファイルに分割し、この分割した複数のファイルに対してのアクセスをそれぞれ管理するもので、当該メモリを用いたシステムの上位者にてファイル配下に第１のキーが予め設定されており、当該メモリを用いたシステムの下位者が新たにキーを設定可能に構成されており、かつ、これらのキーの照合によって上記ファイルへのアクセスを可能とするメモリのアクセス管理方法であって、

上記上位者が設定した第１のキーを、この第１のキーに設定されているキー変更用のアクセス条件を参照して下位者のみが知りえる第２のキーに変更し、上記上位者によるファイル配下のキー創成を拒絶するようにしたことを特徴とするメモリのアクセス管理方法。

【請求項３】　上位のファイルと下位のファイルとからなるツリー構造のファイルへのアクセスを管理するファイルアクセス管理方法において、

上位の管理者が管理する上位のファイルに対してアクセス条件を設定し、

上位の管理者から下位の管理者に対して提供するファイルにアクセス条件を設定し、

上位の管理者から下位の管理者に対する受け渡しのためのトランスポートキーを下位のファイルに設定する場合は、当該ファイルの上位のファイルのアクセス条件を参照し、

この上位のファイルのアクセス条件を満足している場合にトランスポートキーを下位のファイルに設定し、

下位のファイルの配下にファイルを設定する場合は当該下位ファイルのアクセス条件を参照してこのアクセス条件を満足している場合にファイルの設定を行うようにしたことを特徴とするファイルアクセス管理方法。

【請求項４】　前記下位のファイルにはトランスポートキーの設定の可否を示す識別情報が付与されており、この識別情報に基づき該ファイルに対するトランスポートキーの設定を禁止するようにしたことを特徴とする請求項３に記載のファイルアクセス管理方法。

2

【請求項５】　前記下位のファイルに対するトランスポートキーが変更された場合、以後該ファイルに対するトランスポートキーの設定を禁止するように前記識別情報を更新するようにしたことを特徴とする請求項４に記載のファイルアクセス管理方法。

【請求項６】　前記下位のファイルにおけるトランスポートキーの設定の可否を示す識別情報は特殊コマンドに基づき更新するようにしたことを特徴とする請求項４に記載のファイルアクセス管理方法。

【請求項７】　前記下位のファイルへのトランスポートキーの設定はトランスポートキー設定用のコマンドにより行い、前記下位のファイルの配下にファイルを設定する場合はトランスポートキー設定用のコマンドとは異なるコマンドにより行うことを特徴とする請求項３に記載のファイルアクセス管理方法。

【請求項８】　上位のファイルと下位のファイルとからなるツリー構造のファイルを有する携帯可能情報処理装置において、

アクセス条件が設定され上位の管理者が利用するための上位のファイルと、

アクセス条件が設定され上位の管理者から提供され下位の管理者が利用する下位のファイルと、

上位のファイルのアクセス条件を参照してこの上位のファイルのアクセス条件を満足している場合に上位の管理者から下位の管理者に対する受け渡しのためのトランスポートキーを下位のファイルに設定する第１の手段と、

下位ファイルのアクセス条件を参照してこのアクセス条件を満足している場合に下位のファイルの配下にファイルを設定する第２の手段と、

を有することを特徴とする携帯可能情報処理装置。

【請求項９】　前記下位のファイルにはトランスポートキーの設定の可否を示す識別情報の記憶領域を有し、この識別情報に基づき該ファイルに対するトランスポートキーの設定を禁止する手段を有することを特徴とする請求項８に記載の携帯可能情報処理装置。

【請求項１０】　前記下位のファイルに対するトランスポートキーが変更された場合、以後該ファイルに対するトランスポートキーの設定を禁止するように前記識別情報を更新する手段を有することを特徴とする請求項８に記載の携帯可能情報処理装置。

【請求項１１】　前記携帯可能情報記憶媒体は、外部装置から供給されるコマンドに基づき動作し、

前記下位のファイルにおけるトランスポートキーの設定の可否を示す識別情報は外部装置より供給される特殊コマンドに基づき更新する手段を有することを特徴とする請求項８に記載の携帯可能情報処理装置。

【請求項１２】　前記携帯可能情報記憶媒体は、外部装置から供給されるコマンドに基づき動作し、

前記第１の手段による下位のファイルへのトランスポートキーの設定はトランスポートキー設定用のコマンドに

より動作し、前記第２の手段による下位のファイルの配下へのファイルの設定はトランスポートキー設定用のコマンドとは異なるコマンドにより動作することを特徴とする請求項８に記載の携帯可能情報処理装置。

【請求項１３】　上位のファイルと下位のファイルとからなるツリー構造のファイルを創成し、かつ制御手段により各ファイルへのアクセス管理を行う情報処理装置用制御プログラムを記憶した媒体において、

前記制御プログラムは前記制御手段に対して、

上位の管理者によりアクセス条件が設定された上位のファイルを創成させるステップと、

上位の管理者から提供され下位の管理者が利用するためにアクセス条件が設定され下位のファイルを創成させるステップと、

上位のファイルのアクセス条件を参照してこの上位のファイルのアクセス条件を満足している場合に上位の管理者から下位の管理者に対する受け渡しのためのトランスポートキーを下位のファイルに設定させるステップと、

下位ファイルのアクセス条件を参照してこのアクセス条件を満足している場合に下位のファイルの配下にファイルを設定させるステップとから構成されこの制御プログラムを記憶したことを特徴とする情報処理装置用制御プログラムを記憶した媒体。

【請求項１４】　前記情報処理装置用制御プログラムを記憶したプログラムメモリと、前記ファイルが割り当てられるデータメモリと、該ファイルへのアクセス管理を行う制御回路と、外部装置とのインターフェイスが１つのモジュール内に構成されていることを特徴とする請求項１３に記載の情報処理装置用制御プログラムを記憶した媒体。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】本発明は、たとえば、不揮発性メモリ、および、これらを制御するＣＰＵなどの制御素子を有するＩＣチップを内蔵したＩＣカードなどの電子装置において、上記メモリ内に分割設定される複数のファイルに対してのアクセスを管理するメモリのアクセス管理方法に関する。

【０００２】

【従来の技術】最近、携帯可能なデータ記憶媒体として、不揮発性のデータメモリ、および、これらを制御するＣＰＵ（セントラル・プロセッシング・ユニット）などの制御素子を有するＩＣチップを内蔵したＩＣカードが注目されている。

【０００３】この種のＩＣカードは、内蔵するデータメモリを複数のファイルに分割し、かつ、個々のファイルには、利用アプリケーションが運用時に必要なデータなどが格納されるようになっており、外部装置からアプリケーション識別名などを入力することにより、選択的に対応ファイルのみが使用可能な状態を実現するようになっている。このため、複数のアプリケーションデータをファイル分けし、１枚のＩＣカードに格納することにより、多目的利用が可能なようになっている。

【０００４】さて、この多目的を指向するＩＣカードにおいては、カード発行者及びアプリケーション提供者の権限を明確に区分する事が要求される。

【０００５】この権限とは、各当事者に割り当てられている暗証番号が照合された際、ＩＣカードへのアクセスの可否範囲を明確にする事で実現できる。

【０００６】従来のＩＣカードでは、カード発行者がアプリケーション提供者に対してファイルを提供する際、当該ファイルにアプリケーション提供者に対するトランスポートキーを設定する。アプリケーション提供者は、このカードを受け取ると、まず当該トランスポートキーを自身のみが知りえる提供者キーに書き換え、この後、アプリケーション提供者に与えられたファイル内の管理を、この提供者キーにて行えるようになされている。

【０００７】さてここで、ファイル内の管理を行える環境を定義する情報として、一般的に、当該ファイルに付与されたアクセス条件が挙げられる。つまり、カード発行者がアプリケーション提供者に提供するファイルには、アプリケーション提供者キーの照合が必要というアクセス条件が付与されるべきである。

【０００８】上述したファイル内の管理行為の一つとして、アプリケーションにて使用するファイルを設定することが挙げられる。この行為には、先に述べたアプリケーション提供者用トランスポートキーの設定をも含まれていることを考える。

【０００９】この場合、ファイル内の管理をアプリケーション提供者のみにて行わせるように当該ファイルのアクセス条件を設定すると、発行者がアプリケーション提供者用トランスポートキーを設定するために満足しなければならないアプリケーション提供者キーが存在していないため、発行者は、当該トランスポートキーを永久に設定できない事になる。

【００１０】これを回避するために、当該ファイルに与えるアクセス条件として、アプリケーション提供者キーまたは発行者キーのいずれかと設定する方法が考えられる。

【００１１】つまり、発行者が当該ファイルにアプリケーション提供者用トランスポートキーを設定する場合、発行者キーを照合することによりアクセス条件を満足させ、一方、アプリケーション提供者が当該ファイル内の他の管理行為を行う場合には、自身のアプリケーション提供者キーを照合することによりアクセス条件を満足させるようにアクセス条件を設定すれば良い。

【００１２】しかしながら、この場合、当該ファイルに設定されているアクセス条件は発行者とアプリケーション提供者の双方によりアクセス可能となり、当該ファイルへの管理権限は、発行者（上位者）からアプリケーシ

ョン提供者（下位者）に委譲されたとは言えないことになる。

【００１３】

【発明が解決しようとする課題】上記したように、発行者が当該ファイルにアプリケーション提供者用トランスポートキーを設定する場合、発行者キーを照合することによりアクセス条件を満足させ、一方、アプリケーション提供者が当該ファイル内の他の管理行為を行う場合には、自身のアプリケーション提供者キーを照合することによりアクセス条件を満足させた場合、当該ファイルに設定されているアクセス条件により、当該ファイルへの管理権限は、発行者（上位者）からアプリケーション提供者（下位者）に委譲されたとは言えないという問題があった。

【００１４】そこで、この発明は、たとえば、上位者からのデータファイルへの関与をなくし、上位者から下位者へ当該データファイルまたはメモリ領域へのアクセスの権限を委譲することができるメモリのアクセス管理方法を提供することを目的とする。

【００１５】

【課題を解決するための手段】本発明のメモリのアクセス管理方法は、メモリを複数のファイルに分割し、この分割した複数のファイルに対してのアクセスをそれぞれ管理するもので、当該メモリを用いたシステムの上位者にてファイル配下に第１のキーが予め設定されており、当該メモリを用いたシステムの下位者が新たにキーを設定可能に構成されており、かつ、これらのキーの照合によって上記ファイルへのアクセスを可能とするメモリのアクセス管理方法であって、上記上位者が設定した第１のキーを、この第１のキーに設定されているキー変更用のアクセス条件を参照して下位者のみが知りえる第２のキーに変更し、この第２のキーに変更された際、上記上位者によるファイル配下のキー創成を拒絶するようにしたことを特徴とする。

【００１６】本発明のメモリのアクセス管理方法は、メモリを複数のファイルに分割し、この分割した複数のファイルに対してのアクセスをそれぞれ管理するもので、当該メモリを用いたシステムの上位者にてファイル配下に第１のキーが予め設定されており、当該メモリを用いたシステムの下位者が新たにキーを設定可能に構成されており、かつ、これらのキーの照合によって上記ファイルへのアクセスを可能とするメモリのアクセス管理方法であって、上記上位者が設定した第１のキーを、この第１のキーに設定されているキー変更用のアクセス条件を参照して下位者のみが知りえる第２のキーに変更し、上記上位者によるファイル配下のキー創成を拒絶するようにしたことを特徴とする。

【００１７】本発明のファイルアクセス管理方法は、上位のファイルと下位のファイルとからなるツリー構造のファイルへのアクセスを管理するファイルアクセス管理

方法において、上位の管理者が管理する上位のファイルに対してアクセス条件を設定し、上位の管理者から下位の管理者に対して提供するファイルにアクセス条件を設定し、上位の管理者から下位の管理者に対する受け渡しのためのトランスポートキーを下位のファイルに設定する場合は、当該ファイルの上位のファイルのアクセス条件を参照し、この上位のファイルのアクセス条件を満足している場合にトランスポートキーを下位のファイルに設定し、下位のファイルの配下にファイルを設定する場合は当該下位ファイルのアクセス条件を参照してこのアクセス条件を満足している場合にファイルの設定を行うようにしたことを特徴とする。

【００１８】本発明の携帯可能情報処理装置は、上位のファイルと下位のファイルとからなるツリー構造のファイルを有する携帯可能情報処理装置において、アクセス条件が設定され上位の管理者が利用するための上位のファイルと、アクセス条件が設定され上位の管理者から提供され下位の管理者が利用する下位のファイルと、上位のファイルのアクセス条件を参照してこの上位のファイルのアクセス条件を満足している場合に上位の管理者から下位の管理者に対する受け渡しのためのトランスポートキーを下位のファイルに設定する第１の手段と、下位ファイルのアクセス条件を参照してこのアクセス条件を満足している場合に下位のファイルの配下にファイルを設定する第２の手段とから構成されている。

【００１９】本発明の情報処理装置用制御プログラムを記憶した媒体は、上位のファイルと下位のファイルとからなるツリー構造のファイルを創成し、かつ制御手段により各ファイルへのアクセス管理を行う情報処理装置用制御プログラムを記憶した媒体において、前記制御プログラムは前記制御手段に対して、上位の管理者によりアクセス条件が設定された上位のファイルを創成させるステップと、上位の管理者から提供され下位の管理者が利用するためにアクセス条件が設定され下位のファイルを創成させるステップと、上位のファイルのアクセス条件を参照してこの上位のファイルのアクセス条件を満足している場合に上位の管理者から下位の管理者に対する受け渡しのためのトランスポートキーを下位のファイルに設定させるステップと、下位ファイルのアクセス条件を参照してこのアクセス条件を満足している場合に下位のファイルの配下にファイルを設定させるステップとから構成されこの制御プログラムを記憶したことを特徴とする。

【００２０】

【発明の実施の形態】以下、本発明の実施の形態について図面を参照して説明する。

【００２１】図１は、本実施の形態に係る携帯可能電子装置としてのＩＣカードが適用される、たとえば、金融システムあるいはショッピングシステムなどの端末装置として用いられるカード取扱装置の構成例を示すもので

ある。すなわち、この装置は、ＩＣカード１をカードリーダ・ライタ２を介してＣＰＵなどからなる制御部３と接続可能とするとともに、制御部３にキーボード４、ＣＲＴディスプレイ装置５、プリンタ６、および、フロッピーディスク装置７を接続して構成される。

【００２２】図２は、ＩＣカード１の構成例を示すものであり、制御部としての制御素子（たとえば、ＣＰＵ）１１、記憶内容が消去可能な不揮発性のデータメモリ１２、ワーキングメモリ１３、プログラムメモリ１４、および、カードリーダ・ライタ２との電気的接触を得るためのコンタクト部１５によって構成されている。これらのうち、破線内の部分（制御素子１１、データメモリ１２、ワーキングメモリ１３、プログラムメモリ１４）は１つのＩＣチップ（あるいは複数）１０で構成されて、さらに、実開平２－１７３８１号にて知られるように、ＩＣチップ１０とコンタクト部１５とが一体的にＩＣモジュール化されて、ＩＣカード本体内に埋設されている。

【００２３】データメモリ１２は、各種データの記憶に使用され、たとえば、ＥＥＰＲＯＭなどで構成されている。ワーキングメモリ１３は、制御素子１１が処理を行なう際の処理データを一時的に保持するためのメモリであり、たとえば、ＲＡＭなどで構成される。プログラムメモリ１４は、たとえば、マスクＲＯＭで構成されており、制御素子１１のプログラムなどを記憶するものである。

【００２４】データメモリ１２は、たとえば、図３に示すように、制御領域１２０、ディレクトリ１２１、空き領域１２２、および、エリア群１２３に分割されている。エリア群１２３は、複数のデータエリアおよびキーエリアを有することができ、かつ、データファイル（ＤＦ）と呼ばれる概念でグループ化することができる。なお、後述するマスタファイル（ＭＦ）は、データファイルの１つの形態として一括管理される。

【００２５】データファイルは、対応するアプリケーションにて使用されるデータエリア、および、キーエリアを一括して管理するためのファイルである。

【００２６】データエリアは、たとえば、取引データなどのように、必要に応じて読み書きするためのデータを格納するエリアである。

【００２７】キーエリアは、たとえば、暗証番号などの格納に利用されているエリアであり、書込み／書換え／照合の対象になり、読出しはできないようになっている。

【００２８】なお、これらのエリアは、図３に示すように、エリア群１２３として一括して割当てられている。また、これらのファイルあるいはエリアは、データメモリ１２内のディレクトリ１２１を用いることにより、それぞれの物理的位置などを制御素子１１が認識するようになっている。

【００２９】さらに、図３の制御領域１２０には、エリア群１２３の先頭アドレス情報、および、空き領域１２２の先頭アドレス情報が格納されている。

【００３０】図３のディレクトリ１２１は、図４に示すように、各データファイルおよびエリアに対応する各種定義情報が格納される。

【００３１】図４（ａ）は、データファイルの名称を定義する情報である。この定義情報は、ディレクトリ１２１内でデータファイル名定義情報を識別するためのデータＰＴＮ、本データファイルに割当てられたファイル通し番号ＤＦＳＮ、本データファイルの親ファイルの通し番号ＰＦＳＮ、本データファイルに付与されたファイル名ＤＦnameおよびその長さを示すデータＮＬ、および、これらのデータの正当性をチェックするためのデータＢＣＣから構成される。

【００３２】図４（ｂ）は、データファイルの管理情報を定義する情報である。この定義情報は、ディレクトリ１２１内でデータファイル名定義情報を識別するためのデータＰＴＮ、本データファイルに割当てられたファイル通し番号ＤＦＳＮ、本データファイルの親ファイルの通し番号ＰＦＳＮ、データファイルサイズＤＦＳ、本データファイルの付加情報が格納されるデータエリアを識別するためのＡＡＩＤ、当該付加情報を出力するか否かなどを規定するＴＹＰＥ、キーの種別を禁止するＵＣＦ、データファイルのアクセス条件を示すＤＦＡＣ、本データファイルの状態を保持するためのＤＦＳＴ、本データファイルの配下に位置するデータファイルおよびエリアにより使用されているバイト数ＵＳ、および、これらのデータの正当性をチェックするためのデータＢＣＣから構成される。

【００３３】ここでＤＦＳＴの特定ビット（例えば８ビット目）は、該ＤＦのトランスポートキーが変更されたか否かを示すトランスポートビットとして使用される。

【００３４】また、特にＡＡＩＤは、後述するデータファイル選択コマンドにてデータファイルが選択された際に、必要に応じてそれに示されるデータエリアの内容を出力する。

【００３５】図４（ｃ）は、各種取引データなどを格納するエリアを定義する情報である。この定義情報は、ディレクトリ１２１内でエリア定義情報を識別するためのデータＰＴＮ、本エリアが属するデータファイルの通し番号ＤＦＳＮ、エリアに対してアクセスする際の識別番号ＡＩＤ、エリアの先頭アドレスを示すＡＴＯＰ、エリアサイズを示すＡＳＩＺ、エリアのアクセス条件を示すＡＡＣ、エリアの状態を保持するＡＳＴ、および、これらのデータの正当性をチェックするためのデータＢＣＣから構成される。

【００３６】図４（ｄ）は、各種キーデータを格納するエリアを定義する情報である。この定義情報は、ディレクトリ１２１内でキーエリア定義情報を識別するための

データＰＴＮ、本エリアが属するデータファイルの通し番号ＤＦＳＮ、エリアに対してアクセスする際の識別番号ＫＩＤ、エリアの先頭アドレスを示すＫＴＯＰ、エリアサイズを示すＫＳＩＺ、キーの種別を示すＣＦ、キーのアクセス条件を示すＫＡＣ、キーの状態を保持するＫＳＴ、および、これらのデータの正当性をチェックするためのデータＢＣＣから構成される。

【００３７】これらに使用されている識別情報ＰＴＮは、たとえば、１バイトで構成されており、データファイルの名称を定義するもの（図４（ａ））に対しては‘００’が、データファイルの管理情報を定義するもの（図４（ｂ））に対しては‘０１’が、データエリアを定義するもの（図中（ｃ））に対しては‘０２’が、また、キーエリアを定義するもの（図４（ｄ））に対しては‘０３’が、それぞれ使用される。

【００３８】図５は、ファイルの構造例を示している。この図において、ＤＦｎｎはデータファイルを、Ｄｎｎはデータエリアを、Ｋｎｎはキーエリアを、それぞれ示している。

【００３９】図示するように、ＩＣカード１内のメモリ１２において、マスタファイル（ＭＦ）の配下には、データファイルＤＦ１，ＤＦ２が、また、キーエリアＫ００，Ｋ０１、データエリアＤ００，Ｄ０１が、それぞれ設定されている。

【００４０】また、データファイルＤＦ１の配下には、データファイルＤＦ１－１，ＤＦ１－２が、また、キーエリアＫ１１，Ｋ１２、データエリアＤ１１，Ｄ１２が、それぞれ設定されている。

【００４１】また、データファイルＤＦ１－１の配下には、キーエリアＫ１１１，Ｋ１１２、データエリアＤ１１１／Ｄ１１２が、また、データファイルＤＦ１－２の配下には、キーエリアＫ１２１，Ｋ１２２、データエリアＤ１２１，Ｄ１２２が、それぞれ設定されている。

【００４２】一方、データファイルＤＦ２の配下には、データファイルＤＦ２－１，ＤＦ２－２が、また、キーエリアＫ２１，Ｋ２２、データエリアＤ２１，Ｄ２２が、それぞれ設定されている。

【００４３】また、データファイルＤＦ２－１の配下には、キーエリアＫ２１１，Ｋ２１２、データエリアＤ２１１，Ｄ２１２が、また、データファイルＤＦ２－２の配下には、キーエリアＫ２２１，Ｋ２２２、データエリアＤ２２１，Ｄ２２２が、それぞれ設定されている。

【００４４】これらの各種定義情報は、一括して図６に示すように、ディレクトリ１２１に格納される。図示するように、各定義情報には、ＤＦＳＮ（ファイル通し番号）が、ファイル創成時に自動的に付与される。このＤＦＳＮ、および、データファイル定義情報に格納される親ファイルのシーケンス番号により、各ファイルの関連状態を制御素子１１が認識する。

【００４５】たとえば、データファイルＤＦ１－１の定義情報（通し番号＃１３）は、ＤＦＳＮが‘０３’、また、ＰＦＳＮが‘０１’となっている。すなわち、本データファイルは、ファイルシーケンス番号‘０３’が創成時に付与され、同時に本データファイルがＤＦ１の配下に創成されることを認識し、データファイルＤＦ１のＤＦＳＮ（‘０１’）をＰＦＳＮとして付与する。

【００４６】図７は、データファイル（ＤＦ）創成のための動作を説明するフローチヤートを示しており、以下それについて説明する。ＩＣカード１が、外部から入力されるデータファイル創成コマンドを受信すると、まず、使用可能状態、すなわち、カレント状態となっているデータファイル（以後、カレントＤＦと称す）を認識する（ＳＴ１）。特に、ＩＣカード１への電気的活性化の直後は、カレントＤＦはマスタファイル（ＭＦ）となる。

【００４７】カレントＤＦを認識すると、次に、カレントＤＦ定義情報中のアクセス条件情報のうち、ファイル創成に関する情報を参照する。この条件と、後述するＲＡＭ上の照合状態保持領域Ａのみを比較し、アクセス条件が要求しているキーの照合状態が確立されているか否かを判断する（ＳＴ２）。

【００４８】もし確立されていなければ、アクセス条件不一致を意味するレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ３）。また、もし確立されていれば、次に、コマンド内に設定されているデータファイルのファイル名（ＤＦ－ＩＤ）を抽出し、カレントＤＦが有するＦＳＮと同一の値を親ＦＳＮとして有しており、さらに、抽出したファイル名と同一のファイル名を有しているデータファイル定義情報が存在するか否かを確認する（ＳＴ４）。

【００４９】もし存在するとしたら、ＩＤ重複異常を意味するレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ５）。また、もし存在しないとしたら、コマンドにて与えられたデータファイル創成のためのデータにより、図４（ａ）～（ｃ）に示すデータファイル定義情報を生成し（ＳＴ６）、これを所定領域に書込む（ＳＴ７）。

【００５０】この書込みにおいて、書込みが正常に終了しなかった場合（ＳＴ８）、データ書込み異常を意味するレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ９）。また、書込みが正常に終了した場合（ＳＴ８）、正常終了を意味するレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ１０）。

【００５１】図８は、キーエレメンタリファイル（ＥＦ）創成のための動作を説明するフローチヤートを示しており、以下それについて説明する。ＩＣカード１が、外部から入力されるキーＥＦ創成コマンドを受信すると、まず、カレントＤＦを認識する（ＳＴ１１）。

【００５２】カレントＤＦを認識すると、次に、カレントＤＦ定義情報中のアクセス条件情報のうち、ファイル

創成に関する情報を参照する。この条件と、後述するＲＡＭ上の照合状態保持領域Ａのみを比較し、アクセス条件が要求しているキーの照合状態が確立されているか否かを判断する（ＳＴ１２）。

【００５３】もし確立されていなければ、アクセス条件不一致を意味するレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ１３）。また、もし確立されていれば、次に、コマンド電文にて指示されているエレメンタリファイル名（ＥＦ－ＩＤ）を参照し、アクセス対象となっているカレントＤＦ内に、当該エレメンタリファイル名が存在するか否かをチェックする（ＳＴ１４）。もし存在する場合には、ＩＤ重複異常を示すレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ１５）。

【００５４】もし存在しなければ、次に、コマンド電文にて指定されているキーＥＦのサイズデータを参照し、アクセス対象となっているカレントＤＦ内の空き領域サイズと比較する（ＳＴ１６）。この比較においては、指定されたキーＥＦのサイズに当該キーＥＦを創成した際に使用するディレクトリ情報のサイズを加算したものに対して、前記空き領域サイズがそれ以上であるか否かをチェックする。もし前者が後者よりも大きい場合には、指定サイズ異常を示すレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ１７）。

【００５５】もしそうでなければ、次に、コマンド電文にて指定されたキーのタイプと、サイズの正当性をチェックする（ＳＴ１８）。このとき、キーのタイプが「認証関連キー」となっていればサイズが例えば１０バイト、また、キーのタイプが「照合キー」となっていればサイズは例えば３～１８バイトであるとき、サイズが正当であると判断する。もし正当でないと判断された場合には、指定サイズ整合異常を示すレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ１９）。

【００５６】ここで、サイズが正当であったと判断された場合には、受信したコマンドに基づき、ディレクトリに格納すべきキーＥＦ定義情報を生成し（ＳＴ２０）、これを所定領域に書込む（ＳＴ２１）。このとき、ステータス情報の第８ビット目は、コマンド電文にて指定されたキータイプ情報のうちの第１ビット目に依存して、その値が決定される。すなわち、後者のビット値と同様の値を前者のビットに設定する。

【００５７】このステータス情報の第８ビット目は、キーデータの変更がなされたか否かを示すビットであり、当該ビットが「１」の場合、変更行為がなされていないことを示し、また、「０」であればその行為がなされたことを示している。

【００５８】したがって、上記キータイプ情報の第１ビット目が「１」の場合には、キー変更を行なわない限りステータス情報の第８ビット目が「０」とはならず、また、「０」の場合には、キー変更を行なうか否かに関わらず、ステータス情報の第８ビット目が「０」となっている（つまり、書換え行為が暗黙的に行なわれたことと等価になる）。

【００５９】上記キーＥＦ定義情報の書込みにおいて、書込みが正常に終了しなかった場合（ＳＴ２２）、データ書込み異常を意味するレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ２３）。また、書込みが正常に終了した場合（ＳＴ２２）、正常終了を意味するレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ２４）。

【００６０】図９は、トランスポートキーエレメンタリファイル（ＥＦ）を創成する場合の動作を説明するフローチャートを示しており、以下それについて説明する。ＩＣカード１が、外部から入力されるトランスポートＥＦ創成コマンドを受信すると、まず、カレントＤＦを認識する（ＳＴ１０１）。

【００６１】カレントＤＦを認識すると、次に、当該データファイル（ＤＦ）に設定されているステータス情報ＤＦＳＴ中のトランスポートビットがオンになっているか否かをチェックする（ＳＴ１０２）。

【００６２】もしトランスポートビットがオンとなっていれば、当該コマンドは拒絶され、トランスポート不可を意味するレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ１０３）。また、もしトランスポートビットがオフとなっていれば、このトランスポートキーＥＦ創成コマンドが許容される。なお、このトランスポートビットは、データファイル創成時にはオフとなっており、またトランスポートキー自身がキー変更コマンドにより変更された時点でオンとなる。

【００６３】トランスポートキーＥＦ創成コマンドが許容されると、次に、当該カレントＤＦの親データファイル（ＤＦ）のアクセス条件を参照する。この条件と、後述するＲＡＭ上の照合状態保持領域Ａのみを比較し、アクセス条件が要求しているキーの照合状態が確立されているか否かを判断する（ＳＴ１０４）。

【００６４】もし確立されていなければ、アクセス条件不一致を意味するレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ１０５）。また、もし確立されていれば、次に、コマンド電文にて指示されているエレメンタリファイル名（ＥＦ－ＩＤ）を参照し、アクセス対象となっているカレントＤＦ内に、当該エレメンタリファイル名が存在するか否かをチェックする（ＳＴ１０６）。もし存在する場合には、ＩＤ重複異常を示すレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ１０７）。

【００６５】もし存在しなければ、次に、コマンド電文にて指定されているキーＥＦのサイズデータを参照し、アクセス対象となっているカレントＤＦ内の空き領域サイズと比較する（ＳＴ１０８）。この比較においては、指定されたキーＥＦのサイズに当該キーＥＦを創成した際に使用するディレクトリ情報のサイズを加算したもの

に対して、前記空き領域サイズがそれ以上であるか否かをチェックする。もし前者が後者よりも大きい場合には、指定サイズ異常を示すレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ１０９）。

【００６６】もしそうでなければ、次に、コマンド電文にて指定されたキーのタイプと、サイズの正当性をチェックする（ＳＴ１１０）。このとき、キーのタイプが「認証関連キー」となっていればサイズが例えば１０バイト、また、キーのタイプが「照合キー」となっていればサイズは例えば３～１８バイトであるとき、サイズが正当であると判断する。もし正当でないと判断された場合には、指定サイズ整合異常を示すレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ１１１）。

【００６７】ここで、サイズが正当であったと判断された場合には、受信したコマンドに基づき、ディレクトリに格納すべきキーＥＦ定義情報を生成し（ＳＴ１１２）、これを所定領域に書込む（ＳＴ１１３）。このとき、ステータス情報の第８ビット目は、コマンド電文にて指定されたキータイプ情報のうちの第１ビット目に依存して、その値が決定される。すなわち、後者のビット値と同様の値を前者のビットに設定する。

【００６８】このステータス情報の第８ビット目は、キーデータの変更がなされたか否かを示すビットであり、当該ビットが「１」の場合、変更行為がなされていないことを示し、また、「０」であればその行為がなされたことを示している。

【００６９】したがって、上記キータイプ情報の第１ビット目が「１」の場合には、キー変更を行なわない限りステータス情報の第８ビット目が「０」とはならず、また、「０」の場合には、キー変更を行なうか否かに関わらず、ステータス情報の第８ビット目が「０」となっている（つまり、書換え行為が暗黙的に行なわれたことと等価になる）。

【００７０】上記キーＥＦ定義情報の書込みにおいて、書込みが正常に終了しなかった場合（ＳＴ１１４）、データ書込み異常を意味するレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ１１５）。また、書込みが正常に終了した場合（ＳＴ１１４）、正常終了を意味するレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ１１６）。

【００７１】図１０は、キーデータ設定のための動作を説明するフローチヤートを示しており、以下それについて説明する。ＩＣカード１が、外部から入力されるキーデータ設定コマンドを受信すると、まず、カレントＤＦを認識する（ＳＴ３１）。

【００７２】カレントＤＦを認識すると、次に、コマンド電文にて指示されているエレメンタリファイル名（ＥＦ－ＩＤ）を参照し、アクセス対象となっているカレントＤＦ内に、当該エレメンタリファイル名が存在するか否かをチェックする（ＳＴ３２）。もし存在しない場合には、該当キーＩＤ無しを意味するレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ３３）。

【００７３】もし存在すれば、次に当該キーＥＦ定義情報中のアクセス条件情報のうち、キーデータ設定に関する情報を参照する。この条件と、後述するＲＡＭ上の照合状態保持領域Ａのみを比較し、アクセス条件が要求しているキーの照合状態が確立されているか否かを判断する（ＳＴ３４）。

【００７４】もし確立されていなければ、アクセス条件不一致を意味するレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ３５）。また、もし確立されていれば、次に、対応するキーＥＦ領域内にキーデータが存在するか否かを確認する（ＳＴ３６）。もし存在すれば、既存キーデータ有りを意味するレスポンスデータを出力し、コマンド待ち状態に戻る（ＳＴ３７）。

【００７５】もし存在しなければ、コマンド電文にて指定されたキーのタイプと、入力キーデータのサイズの正当性をチェックする（ＳＴ３８）。このとき、キーのタイプが「認証関連キー」となっていればサイズが例えば８バイト、また、キーのタイプが「照合キー」となっていればサイズは例えば１～１６バイトであるとき、サイズが正当であると判断する。もし正当でないと判断された場合には、入力キーデータサイズ異常を示すレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ３９）。

【００７６】ここで、サイズが正当であったと判断された場合には、次に、当該キーＥＦ定義情報中にて定義されているサイズと、入力されたキーデータのサイズとの比較を行なう（ＳＴ４０）。後者のサイズに例えば「２」を加えたものが、前者のサイズよりも大きい場合には、領域サイズ不足を意味するレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ４１）。

【００７７】そうでなければ、受信したコマンドにて入力されたキーデータに、１バイトの長さの情報および１バイトのＢＣＣを付加して、これを当該キーＥＦ領域に格納し（ＳＴ４２）、処理結果をレスポンス電文にて出力し、コマンド待ち状態に戻る（ＳＴ４３）。

【００７８】図１１は、キーデータ変更のための動作を説明するフローチヤートを示しており、以下それについて説明する。ＩＣカード１が、外部から入力されるキーデータ変更コマンドを受信すると、まず、カレントＤＦを認識する（ＳＴ５１）。

【００７９】カレントＤＦを認識すると、次に、コマンド電文にて指示されているエレメンタリファイル名（ＥＦ－ＩＤ）を参照し、アクセス対象となっているカレントＤＦ内に、当該エレメンタリファイル名が存在するか否かをチェックする（ＳＴ５２）。もし存在しない場合には、該当キーＩＤ無しを意味するレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ５３）。

【００８０】もし存在すれば、次に当該キーＥＦ定義情

報中のアクセス条件情報のうち、キーデータ変更に関する情報を参照する。この条件と、後述するＲＡＭ上の照合状態保持領域ＡおよびＢを比較し、アクセス条件が要求しているキーの照合状態が確立されているか否かを判断する（ＳＴ５４）。

【００８１】もし確立されていなければ、アクセス条件不一致を意味するレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ５５）。また、もし確立されていれば、次に、対応するキーＥＦ領域内にキーデータが存在するか否かを確認する（ＳＴ５６）。もし存在しなければ、既存キーデータ無しを意味するレスポンスデータを出力し、コマンド待ち状態に戻る（ＳＴ５７）。

【００８２】もし存在すれば、コマンド電文にて指定されたキーのタイプと、入力キーデータのサイズの正当性をチェックする（ＳＴ５８）。このとき、キーのタイプが「認証関連キー」となっていればサイズが例えば８バイト、また、キーのタイプが「照合キー」となっていればサイズは例えば１～１６バイトであるとき、サイズが正当であると判断する。もし正当でないと判断された場合には、入力キーデータサイズ異常を示すレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ５９）。

【００８３】ここで、サイズが正当であったと判断された場合には、次に、当該キーＥＦ定義情報中にて定義されているサイズと、入力されたキーデータのサイズとの比較を行なう（ＳＴ６０）。後者のサイズに例えば「２」を加えたものが、前者のサイズよりも大きい場合には、領域サイズ不足を意味するレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ６１）。

【００８４】そうでなければ、受信したコマンドにて入力されたキーデータに、１バイトの長さの情報および１バイトのＢＣＣを付加して、これを当該キーＥＦ領域に格納し（ＳＴ６２）、その処理結果をレスポンス電文にて出力し、コマンド待ち状態に戻る（ＳＴ６３）。また、このとき、キーＥＦ定義情報にあるステータス情報の第８ビット目を「０」とする。

【００８５】図１２は、キー照合のための動作を説明するフローチヤートを示しており、以下それについて説明する。ＩＣカード１が、外部から入力されるキー照合コマンドを受信すると、まず、カレントＤＦを認識する（ＳＴ７１）。

【００８６】カレントＤＦを認識すると、次に、ディレクトリ１２１を検索することにより、カレントＤＦ内に指定されたファイル名（ＩＤ）を有するキーＥＦ定義情報が存在するか否かを確認する（ＳＴ７２）。もし存在しない場合には、該当キーＩＤ無しを示すレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ７３）。

【００８７】もし存在していた場合には、当該キーがロック状態になっているか否かを確認する（ＳＴ７４）。このとき、ロック状態であると判断した場合には、キーロックを示すレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ７５）。

【００８８】もしそうでなければ、コマンド電文内のキーデータと、当該キーＥＦ内に格納されているキーデータとを照合する（ＳＴ７６）。このとき、両者が一致している場合には（ＳＴ７７）、当該キーＥＦ定義情報中の照合ビット指定情報を参照し、所定のＲＡＭ領域の当該情報にて指定されているビット位置を「１」にする（ＳＴ７８）。次に、当該キーＥＦ定義情報中のキー固有の照合不一致カウンタをクリアし（ＳＴ７９）、正常終了を示すレスポンス電文を出力して、コマンド待ち状態に戻る（ＳＴ８０）。

【００８９】なお、所定のＲＡＭ領域は、照合状態保持領域ＡおよびＢに分割されている。どちらの領域の対応ビットを「１」にするかは、当該キーＥＦ定義情報中のキーのステータス情報の第８ビット目の値に依存する。このビットは、当該キーＥＦ定義情報により定義づけられているキーに対し、変更処理が行なわれたか否かを示すものであり、後述するように、「０」となっていれば変更されたキーであり、「１」となっていれば変更処理が行なわれていないキーであることを示す。さらに、これが「０」となっていた場合には、前記照合状態保持領域Ａの、また、「１」となっていれば、前記照合状態保持領域Ｂの、対応ビットを設定することになる。

【００９０】また、キー照合処理において、不一致であると判断した場合には（ＳＴ７７）、まず、当該キーＥＦ定義情報中の照合ビット指定情報、および、ステータス情報を参照し、上記と同様の手順にしたがって、照合状態保持領域ＡまたはＢのいずれかの領域の所定ビットを「０」にする（ＳＴ８１）。

【００９１】次に、キー固有の照合不一致カウンタを１つだけインクリメントする（ＳＴ８２）。このとき、キーＥＦ定義情報中のカウント最大値に達していない場合には（ＳＴ８３）、照合不一致を示すレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ８４）。また、最大値に達していたならば、キーロック済みを示すレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ８５）。

【００９２】照合状態保持領域Ａは、ファイル創成時、キーＥＦ創成時、キーデータ創成時、キーデータ変更時に参照され、照合状態保持領域Ｂはキーデータ変更時に参照される。

【００９３】上述したように、キーが変更済みの場合照合状態保持領域Ａが、未変更であれば照合状態保持領域Ｂのビットがオンされるので、ファイル創成時、キーＥＦ創成時、キーデータ創成時にはキーが変更済みでなければならない。又、キーの変更時には照合状態保持領域Ａ及びＢを参照するのでキーが変更済みであっても未変更であってもキーの変更は可能となる。

【００９４】図１３は、データＥＦへのアクセスのための動作を説明するフローチヤートを示しており、以下そ

れについて説明する。ＩＣカード１が、外部から入力されるデータＥＦアクセスコマンドを受信すると、まず、カレントＤＦを認識する（ＳＴ９１）。

【００９５】カレントＤＦを認識すると、次に、コマンド電文にて指示されているエレメンタリファイル名（ＥＦ－ＩＤ）を参照し、アクセス対象となっているカレントＤＦ内に、当該エレメンタリファイル名が存在するか否かをチェックする（ＳＴ９２）。もし存在しない場合には、該当キーＩＤ無しを意味するレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ９３）。

【００９６】もし存在すれば、次に当該データＥＦ定義情報中のアクセス条件情報のうち、アクセスのタイプ（データ読出し／書込み／変更）に対応するアクセス条件情報を参照する。この条件と、後述するＲＡＭ上の照合状態保持領域Ａを比較し、アクセス条件が要求しているキーの照合状態が確立されているか否かを判断する（ＳＴ９４）。

【００９７】もし確立されていなければ、アクセス条件不一致を意味するレスポンス電文を出力し、コマンド待ち状態に戻る（ＳＴ９５）。もし確立されていれば、次に、対応するデータＥＦ領域内に対してアクセスを行ない（ＳＴ９６）、その処理結果をレスポンス電文として出力し、コマンド待ち状態に戻る（ＳＴ９７）。

【００９８】次に、このような構成において、ＩＣカードに対して各ファイルを創成する手順を説明する。

【００９９】まず、ＩＣカードを製造する際に、カード発行者に渡すべきトランスポートキーのエレメンタリファイルを設定（創生）するために、図９のフローチャートに従って図１４に示すようにマスタファイル（ＭＦ）配下に“キーＥＦ－ａ”を創成し、ここにトランスポートキーを設定する。

【０１００】続いて当該キーエレメンタリファイルＥＦ－ａへの中の設定が図１０のフローチャートに従って設定され、キーの設定に際しては、当該キーＥＦ－ａに付与されているキー設定用のアクセス条件が参照される。

【０１０１】この状態で発行者がカードを受け取ると、図１１のフローチャートに従ってトランスポートキーを変更する。すなわち図１５に示すように発行者は、製造者が設定したトランスポートキーを自身のみが知りえるキー（ａ′）に変更する。この変更には、当該キーＥＦに設定されているキー変更用のアクセス条件を参照する。この時点で、マスタファイル（ＭＦ）に付与されているトランスポートビットがオンされ、以降、製造者によるマスタファイル（ＭＦ）配下のキーの創成が拒絶される。

【０１０２】つぎに、発行者は、自身が必要とするデータＥＦ－ａをマスタファイル（ＭＦ）配下に創成する。この場合、マスタファイル（ＭＦ）に設定されているＥＦ創成用アクセス条件を参照する。

＊【０１０３】また発行者は、アプリケーション提供者に開放するためのデータファイル（ＤＦ）を、マスタファイル（ＭＦ）配下に創成する。ＤＦの設定は図７のフローに従って行われ、この場合、マスタファイル（ＭＦ）に設定されているデータファイル（ＤＦ）創成用アクセス条件を参照する。

【０１０４】つぎに、発行者は当該データファイル（ＤＦ）配下に、アプリケーション提供者に渡すべきトランスポートキーＥＦ－ｂを創成する。トランスポートキーＥＦ－ｂの創成は図９のフローに従って行われ、この場合、参照するアクセス条件は当該データファイル（ＤＦ）に付与されているものではなく、その親ファイルであるマスタファイル（ＭＦ）に付与されているアクセス条件を参照する。

【０１０５】つぎに発行者は、このトランスポートキーＥＦ－ｂ内に、トランスポートキーを設定する。トランスポートキーの設定は図１０のフローに従って行われ、この場合、対象となっているキーＥＦ－ｂに付与されている、キー設定用のアクセス条件を参照する。

【０１０６】この状態でアプリケーション提供者がカードを受け取ると、図１６に示すようにアプリケーション提供者は、発行者が設定したトランスポートキーを自身のみが知りえるキー（ｂ′）に変更する。この変更には、当該キーＥＦに設定されているキー変更用のアクセス条件を参照する。この時点で、データファイル（ＤＦ）に付与されているトランスポートビットがオンされ、以降、発行者によるデータファイル（ＤＦ）配下のキーの創成が拒絶される。

【０１０７】つぎに、アプリケーション提供者は、自身が必要とするデータＥＦ－ｂをデータファイル（ＤＦ）配下に創成する。データＥＦ－ｂの創成は図７のフローに従って行われ、この場合、データファイル（ＤＦ）に設定されているＥＦ創成用アクセス条件を参照する。

【０１０８】またアプリケーション提供者は、自身が必要とするキーＥＦ－ｃをデータファイル（ＤＦ）配下に創成する。キーＥＦ－ｃの創成は図８のフローに従って行われ、この場合も、データファイル（ＤＦ）に設定されているＥＦ創成用アクセス条件を参照する。

【０１０９】コマンドが下記のようになっており、特に、通常のキーＥＦ創成コマンドとトランスポートキーＥＦ創成コマンドとは別のコマンドコード形式となっていて、ＩＣカードはこのコマンドコードによりコマンドの内容を識別している。

【０１１０】すなわち、通常のキーＥＦ創成コマンドとトランスポートキーＥＦ創成コマンドとを識別し、トランスポートキーＥＦ創成コマンドの場合は図９のフローチャートにより処理され、通常のキーＥＦ創成コマンドの場合は図８のフローチャートにより処理される。

＊【０１１１】

エリア（エレメンタリファイル）創成コマンド

19　　20

コマンドコードA／ＡＩＤ（ＥＦ－ＩＤ）／ＡＳＩＺ／ＡＡＣ

キーＥＦ創成コマンド

コマンドコードB／ＫＩＤ／ＫＳＩＺ／ＣＦ／ＡＡＣ

トランスポートキーＥＦ創成コマンド

コマンドコードC／ＫＩＤ／ＫＳＩＺ／ＣＦ／ＡＡＣ

上述したカレントＤＦ配下にトランスポートキーＥＦを創成する場合はカレントＤＦの親ＤＦのアクセス条件を参照する。親ＤＦのアクセス条件は上位者が設定したものであり、トランスポートキーＥＦの創成は上位者が行うことができる。

【０１１２】また、トランスポートキーＥＦへのトランスポートキーの設定時は当該ＥＦのアクセス条件を参照するが、当該ＥＦのアクセス条件はトランスポートキーＥＦの創成時に上位者が設定するものであり、上位者はトランスポートキーＥＦへのトランスポートキーの設定が可能となるように当該ＥＦのアクセス条件を設定する。

【０１１３】カレントＤＦ配下にアプリケーション提供者がキーＥＦを設定する場合はカレントＤＦのキーＥＦ創成用のアクセス条件が参照されるが、ＤＦのアクセス条件はＤＦ創成時に上位者が設定することとなる。

【０１１４】しかし、トランスポートキーの変更後はこのＤＦのアクセス条件を用いる処理（データＥＦ、キーＥＦの創成）はアプリケーション提供者しかおこなうことはできないので、アクセス権の区分は明確となる。

【０１１５】さらに、キーＥＦ創成用のアクセス条件として上位者がトランスポートキーの変更後のキーを設定しておくことでアプリケーション提供者はトランスポートキーを変更した後に自身で用いるキーＥＦを設定し、この自身で用いるキーＥＦのキーの定義情報中の照合ビット指定情報によりＤＦのアクセス条件を満足できる環境を用意し、そして、自身のキーを照合するとデータＥＦの創成のためのＤＦのアクセス条件が満足されて、データＥＦの創成が可能となるようにしておく。

【０１１６】このようにすることで、上位者はＤＦのアクセス条件は設定するものの、その内容については関知することなく、アプリケーション提供者のみがＤＦのアクセス条件を満足できるようになる。

【０１１７】以上説明したように上記発明の実施の形態によれば、データファイル（ＤＦ）配下にＥＦ（またはＤＦ）を創成するために必要となるキーは、当該データファイル（ＤＦ）に付与されているアクセス条件により決定され、またこのアクセス条件は、上位のキー（この実施例においては、発行者が、アプリケーション提供者にとっての上位者になる）を指定することなく実現することができる。

【０１１８】従って、アプリケーション提供者が自身で管理したいファイル（またはメモリ領域）は、自身のキーのみで保護することができる。

【０１１９】これにより、発行者の当該データファイル（ＤＦ）への関与がなされなくなり、従って、発行者からアプリケーション提供者へ、当該ファイル（またはメモリ領域）へのアクセスの権限が委譲されたことになる。

【０１２０】なお、本実施例にあるトランスポートビットは、データファイル（ＤＦ）創成時にはオフとなっており、トランスポートキー自身がキー変更コマンドにより変更された時点でオンとなる旨を記載しているが、キー変更コマンドと連動することなく、例えばトランスポートビットをオンするコマンドにて実施してもよい。この場合、当該コマンドを実行する際に使用されるアクセス条件は、対象とするファイルに設定されているものが参照される。

【０１２１】また、前記実施の形態では、メモリのアクセス管理を行なう電子機器としてＩＣカードを例示したが、これに限定されることなく、メモリのアクセス管理を必要とするメモリを備えた電子機器であれば適用可能である。

【０１２２】

【発明の効果】以上詳述したように本発明によれば、たとえば、上位者からのデータファイルへの関与をなくし、上位者から下位者へ当該データファイルまたはメモリ領域へのアクセスの権限を委譲することができるメモリのアクセス管理方法を提供できる。

【図面の簡単な説明】

【図１】本発明の実施の形態に係るＩＣカードが適用されるカード取扱装置の構成例を示すブロック図。

【図２】ＩＣカードの構成例を示すブロック図。

【図３】データメモリの構成例を示すメモリマップ図。

【図４】各種定義情報のフォーマット例を示す図。

【図５】データメモリ内に設定されるファイルの構造例を示す図。

【図６】データメモリ内に設定されるディレクトリの構成例を示す図。

【図７】データファイル創成のための動作を説明するフローチャート。

【図８】キーエレメンタリファイル創成のための動作を説明するフローチャート。

【図９】トランスポートキーエレメンタリファイル創成のための動作を説明するフローチャート。

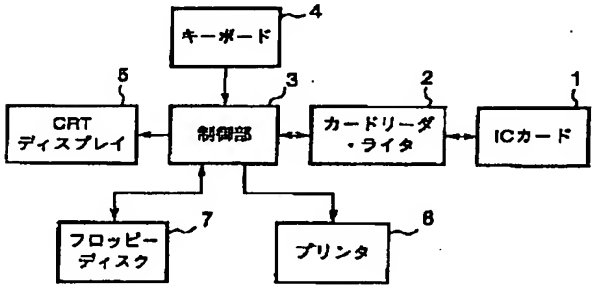
【図１０】キーデータ設定のための動作を説明するフローチャート。

【図１１】キーデータ変更のための動作を説明するフローチャート。

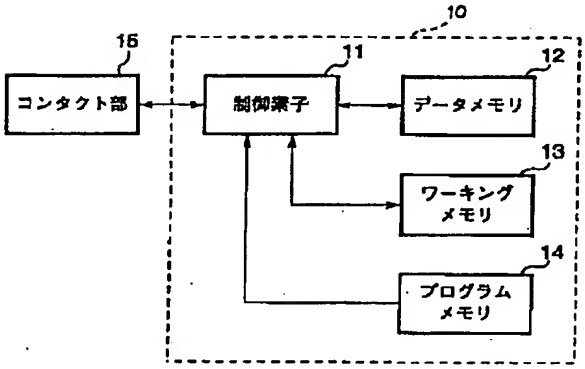
【図１２】キー照合のための動作を説明するフローチャ

ート。
【図１３】データエレメンタリファイルへのアクセスのための動作を説明するフローチャート。
【図１４】製造者によるＩＣカードに対するファイル創成を説明するための図。
【図１５】発行者によるＩＣカードに対するファイル創成を説明するための図。
【図１６】アプリケーション提供者によるＩＣカードに対するファイル創成を説明するための図。
【符号の説明】
１…ＩＣカード
２…カードリーダ・ライタ
３…制御部
４…キーボード
５…ＣＲＴディスプレイ装置
１１…制御素子
１２…データメモリ
１３…ワーキングメモリ
１４…プログラムメモリ
１５…コンタクト部
１２０…制御領域
１２１…ディレクトリ
１２３…エリア群

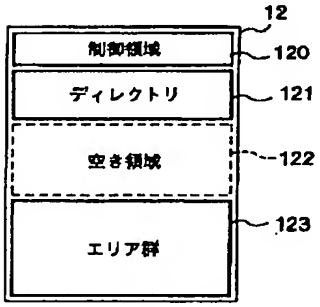
【図１】



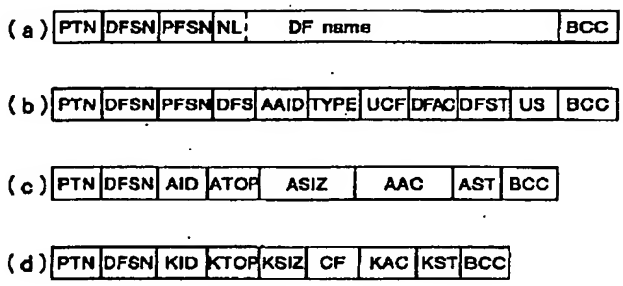
【図２】



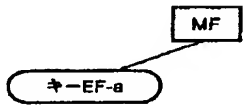
【図３】



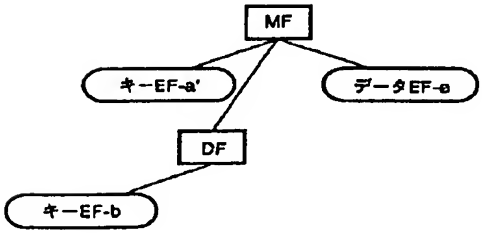
【図４】



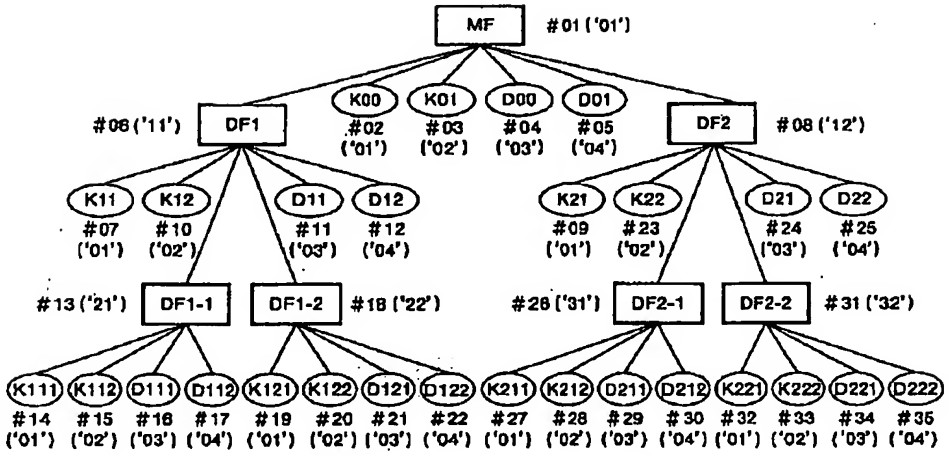
【図１４】



【図１５】



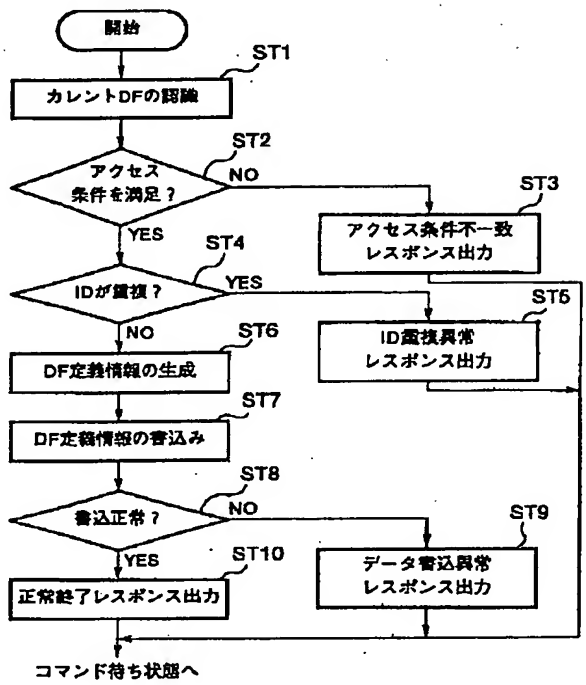
【図５】



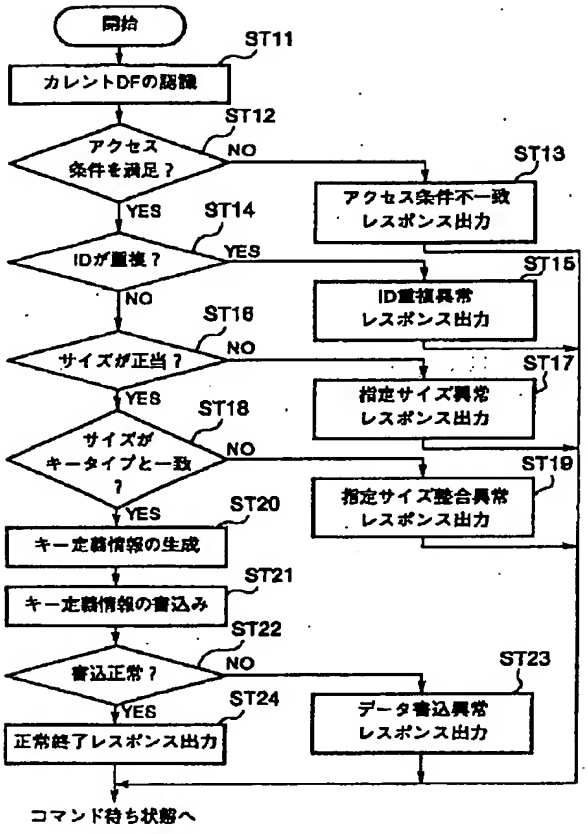
【図６】

通し番号	DFSN	PFSN	FID	
#01	00	00	00	MF 定義情報
#02	00	-	01	K00 定義情報
#03	00	-	02	K01 定義情報
#04	00	-	03	D00 定義情報
#05	00	-	04	D01 定義情報
#06	01	00	11	DF1 定義情報
#07	01	-	01	K11 定義情報
#08	02	00	12	DF2 定義情報
#09	02	-	01	K21 定義情報
#10	01	-	02	K12 定義情報
#11	01	-	03	D11 定義情報
#12	01	-	04	D12 定義情報
#13	03	01	21	DF1-1 定義情報
#14	03	-	01	K111 定義情報
#15	03	-	02	K112 定義情報
#16	03	-	03	D111 定義情報
#17	03	-	04	D112 定義情報
#18	04	01	22	DF1-2 定義情報
#19	04	-	01	K121 定義情報
#20	04	-	02	K122 定義情報
#21	04	-	03	D121 定義情報
#22	04	-	04	D122 定義情報
#23	02	-	02	K22 定義情報
#24	02	-	03	D21 定義情報
#25	02	-	04	D22 定義情報
#26	05	02	31	DF2-1 定義情報
#27	05	-	01	K211 定義情報
#28	05	-	02	K212 定義情報
#29	05	-	03	D211 定義情報
#30	05	-	04	D212 定義情報
#31	06	02	32	DF2-2 定義情報
#32	06	-	01	K221 定義情報
#33	06	-	02	K222 定義情報
#34	06	-	03	D221 定義情報
#35	06	-	04	D222 定義情報

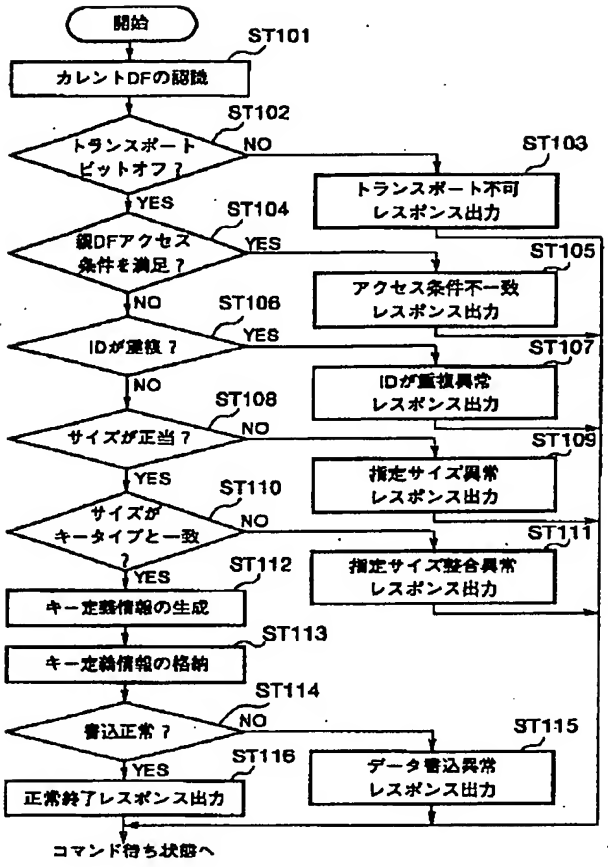
【図７】



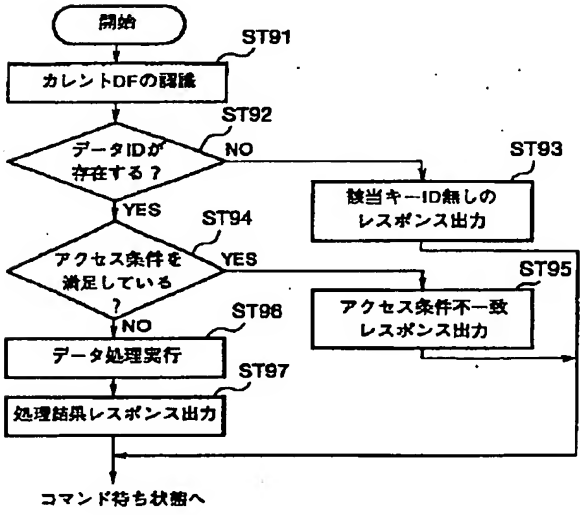
【図８】


開始
ST11
カレントDFの認識
ST12
アクセス条件を満足？
NO
ST13
アクセス条件不一致レスポンス出力
YES
ST14
IDが重複？
YES
ST15
ID重複異常レスポンス出力
NO
ST16
サイズが正当？
NO
ST17
指定サイズ異常レスポンス出力
YES
ST18
サイズがキータイプと一致？
NO
ST19
指定サイズ整合異常レスポンス出力
YES
ST20
キー定義情報の生成
ST21
キー定義情報の書込み
ST22
書込正常？
NO
ST23
データ書込異常レスポンス出力
YES
ST24
正常終了レスポンス出力
コマンド待ち状態へ

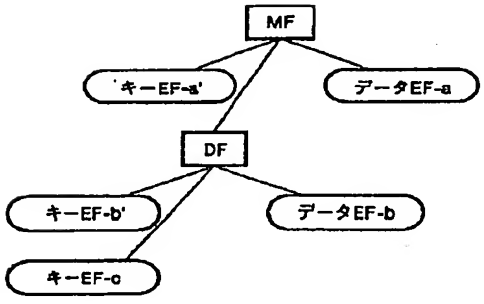
【図９】


開始
ST101
カレントDFの認識
ST102
トランスポートビットオフ？
NO
ST103
トランスポート不可レスポンス出力
YES
ST104
親DFアクセス条件を満足？
YES
ST105
アクセス条件不一致レスポンス出力
NO
ST106
IDが重複？
YES
ST107
IDが重複異常レスポンス出力
NO
ST108
サイズが正当？
NO
ST109
指定サイズ異常レスポンス出力
YES
ST110
サイズがキータイプと一致？
NO
ST111
指定サイズ整合異常レスポンス出力
YES
ST112
キー定義情報の生成
ST113
キー定義情報の格納
ST114
書込正常？
NO
ST115
データ書込異常レスポンス出力
YES
ST116
正常終了レスポンス出力
コマンド待ち状態へ

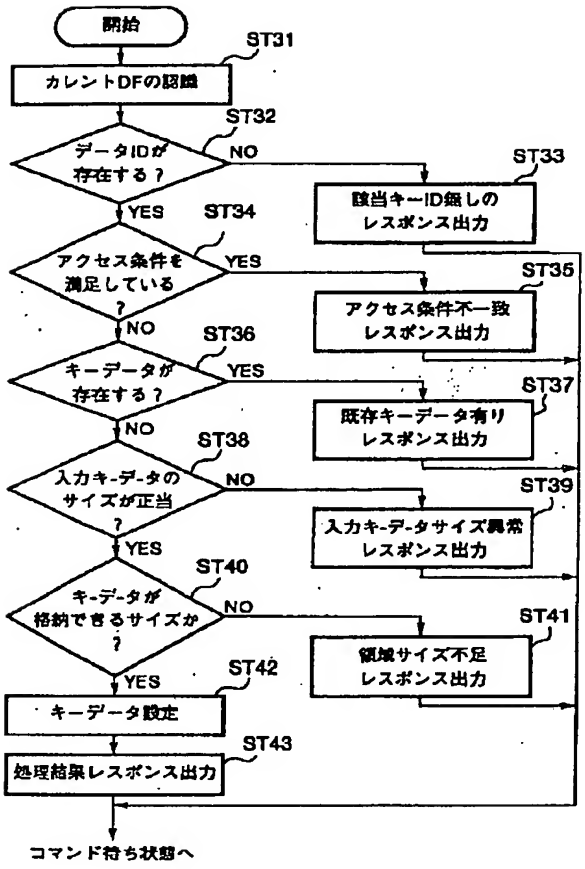
【図１３】


開始
ST91
カレントDFの認識
ST92
データIDが存在する？
NO
ST93
該当キーID無しのレスポンス出力
YES
ST94
アクセス条件を満足している？
YES
ST95
アクセス条件不一致レスポンス出力
NO
ST96
データ処理実行
ST97
処理結果レスポンス出力
コマンド待ち状態へ

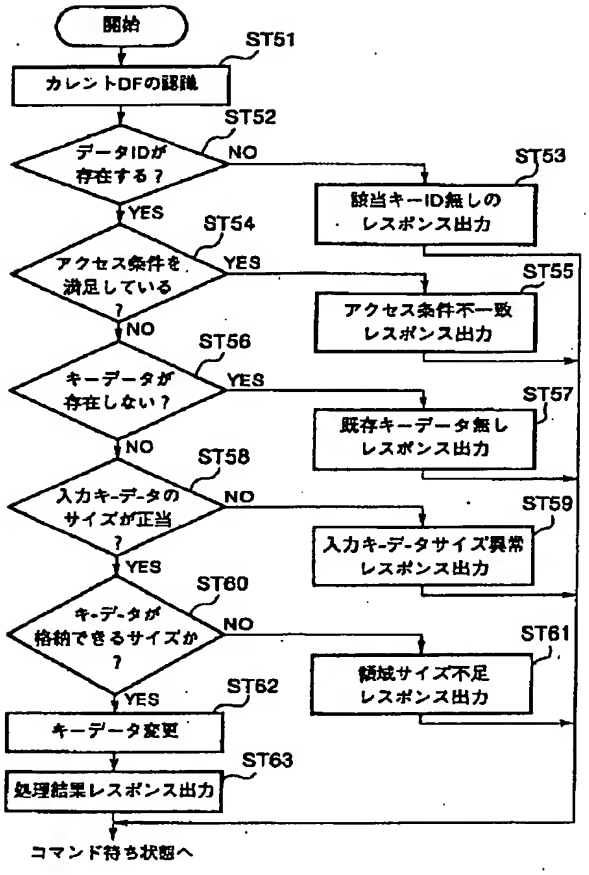
【図１６】


MF
キーEF-a'
データEF-a
DF
キーEF-b'
データEF-b
キーEF-c

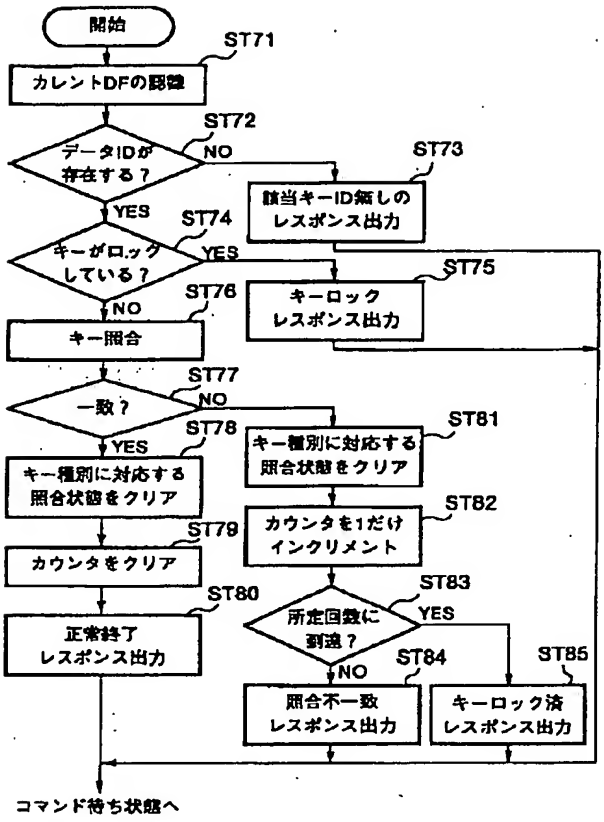
【図１０】


開始
ST31
カレントDFの認識
ST32
データIDが存在する？
NO
ST33
該当キーID無しのレスポンス出力
YES
ST34
アクセス条件を満足している？
YES
ST35
アクセス条件不一致レスポンス出力
NO
ST36
キーデータが存在する？
YES
ST37
既存キーデータ有りレスポンス出力
NO
ST38
入力キーデータのサイズが正当？
NO
ST39
入力キーデータサイズ異常レスポンス出力
YES
ST40
キーデータが格納できるサイズか？
NO
ST41
領域サイズ不足レスポンス出力
YES
ST42
キーデータ設定
ST43
処理結果レスポンス出力
コマンド待ち状態へ

【図１１】


開始
ST51
カレントDFの認識
ST52
データIDが存在する？
NO
ST53
該当キーID無しのレスポンス出力
YES
ST54
アクセス条件を満足している？
YES
ST55
アクセス条件不一致レスポンス出力
NO
ST56
キーデータが存在しない？
YES
ST57
既存キーデータ無しレスポンス出力
NO
ST58
入力キーデータのサイズが正当？
NO
ST59
入力キーデータサイズ異常レスポンス出力
YES
ST60
キーデータが格納できるサイズか？
NO
ST61
領域サイズ不足レスポンス出力
YES
ST62
キーデータ変更
ST63
処理結果レスポンス出力
コマンド待ち状態へ

【図１２】


開始
ST71
カレントDFの取得
ST72
データIDが存在する？
NO
ST73
該当キーID無しのレスポンス出力
YES
ST74
キーがロックしている？
YES
ST75
キーロックレスポンス出力
NO
ST76
キー照合
ST77
一致？
NO
ST81
キー種別に対応する照合状態をクリア
YES
ST78
キー種別に対応する照合状態をクリア
ST82
カウンタを1だけインクリメント
ST79
カウンタをクリア
ST83
所定回数に到達？
YES
ST80
正常終了レスポンス出力
NO
ST84
照合不一致レスポンス出力
ST85
キーロック済レスポンス出力
コマンド待ち状態へ